This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims

1.   (Currently Amended) A method for interfacing with device hardware supporting a plurality of devices included in the device hardware, comprising:

initializing a device interface driver to represent the device hardware as a virtual bus to an operating system and to represent to the operating system each device supported in the device hardware as a device attached to the virtual bus;

initializing the device hardware;

accessing the device hardware to determine the devices supported by included in the device hardware;

generating one device object for each determined device supported by in the device hardware, wherein each generated device object represents the determined device to the operating system; and

reporting the determined devices to the operating system, wherein the operating system loads a device driver for each of the reported devices supported by the device hardware.

2.   (Original) The method of claim 1, further comprising:

reporting to the operating system that the determined devices are dependent on the virtual bus, wherein in response to being notified that the determined devices and virtual bus are dependent, the operating system will not remove the device interface driver representing the virtual bus until the device drivers associated with the determined devices are removed.

3.   (Original) The method of claim 1, further comprising:

reporting to the operating system that a power state of the virtual bus represented by the device interface driver cannot be altered until all the device drivers representing devices attached to the virtual bus have their power state similarly altered.

4. (Original) The method of claim 1, wherein the hardware device comprises a network adaptor and wherein each device available in the network adaptor supports a protocol engine for different communication protocols.

5. (Original) The method of claim 4, wherein each protocol engine processes packets according to a communication protocol and a network protocol, wherein each transport engine supports a different communication protocol and uses a same network protocol.

6. (Original) The method of claim 1, further comprising:

receiving a packet from one device driver;

determining a device queue in the device hardware queuing packets for the device supported by the device hardware corresponding to the device driver; and

writing the received packet to the determined queue.

7. (Original) The method of claim 1, further comprising:

receiving notification from the device hardware concerning transmission of one packet;

determining the device driver for the device in the network adaptor that processed the packet; and

transmitting notification to the determined device driver indicating the notification received from the device hardware.

8. (Original) The method of claim 1, further comprising:

receiving indication of a packet provided by the device hardware;

determining the device driver for the device supported by the network adaptor that processed the provided packet;

invoking a call to cause the determined device driver to process the provided packet.

9. (Original) The method of claim 1, wherein all the device drivers access devices supported by the device hardware through the device interface driver.

10.   (Currently Amended) The method of claim 1, wherein the devices determined ~~to be available~~ in the device hardware for which device objects are generated comprise less than all the devices ~~supported by~~ included in the device hardware.

11.   (Original) The method of claim 1, further comprising:

in response to detecting a change in the configuration of devices supported by the device hardware, signaling the operating system of the changed configuration of devices available in the device hardware, wherein the operating system is capable of loading or unloading device drivers to support the changed configuration of devices available in the device hardware.

12.   (Original) The method of claim 1, wherein the operations of initializing the device hardware, accessing the device hardware to determine devices supported by the device hardware, generating the device objects, and reporting the determined devices to the operating system are performed by the device interface driver.

13.   (Currently Amended) The method of claim 1, wherein one device includes a protocol engine supporting the Internet Small Computer Interface (iSCSI) communication protocol and Transmission Control Protocol (TCP) communication protocol, wherein one device includes a protocol engine supporting an offloaded Local Area Network (LAN) communication protocol, and wherein one device includes a protocol engine supporting non-offloaded-LAN protocol.

14.   (Currently Amended) A system, comprising:

a processor;

an operating system executed by the processor;

a network adaptor supporting a plurality of devices included in the network adaptor;

a device interface driver that represents the network adaptor as a virtual bus to the operating system, and that represents to the operating system each device supported in the device hardware as a device attached to the virtual bus, wherein the processor, when executing the device interface driver, is enabled to:

[[(i)]] access the network adaptor to determine the devices ~~supported~~ included in by the network adaptor;

[[(ii)]] generate one device object for each determined device ~~supported by~~ in the network adaptor, wherein each generated device object represents the determined device to the operating system; and

[[(iii)]] report the determined devices to the operating system, wherein the operating system loads a device driver for each of the reported devices supported by the network adaptor, and wherein to the operating system each determined device is considered attached to the virtual bus.

15.   (Original) The system of claim 14, wherein the processor, when executing the device interface driver, is further enabled to:

report to the operating system that the determined devices are dependent on the virtual bus, wherein in response to being notified that the determined devices and virtual bus are dependent, the operating system will not remove the device interface driver representing the virtual bus until the device drivers associated with the determined devices are removed.

16.   (Original) The system of claim 14, wherein the processor, when executing the device interface driver, is further enabled to:

report to the operating system that a power state of the virtual bus represented by the device interface driver cannot be altered until all the device drivers representing devices attached to the virtual bus have their power state similarly altered.

17.   (Original) The system of claim 14, wherein each device available in the network adaptor supports a protocol engine for different communication protocols.

18.   (Original) The system of claim 17, wherein each protocol engine processes packets according to a communication protocol, wherein each protocol engine supports a different communication protocol and uses a same network protocol.

19.   (Original) The system of claim 14, wherein the processor, when executing the device interface driver, is further enabled to:

receive a packet from one device driver;

determine a device queue in the network adaptor queuing packets for the device supported by the hardware device corresponding to the device driver; and

write the received packet to the determined queue.

20.   (Original) The system of claim 14, wherein the processor, when executing the device interface driver, is further enabled to:

receive notification from the network adaptor concerning transmission of one packet;

determine the device driver for the device in the network adaptor that processed the packet; and

transmit notification to the determined device driver indicating the notification received from the device hardware.

21.   (Original) The system of claim 14, wherein the processor, when executing the device interface driver, is further enabled to:

receive indication of a packet provided by the network adaptor;

determine the device driver for the device supported by the network adaptor that processed the provided packet;

invoke a call to cause the determined device driver to process the provided packet.

22.   (Original) The system of claim 14, wherein all the device drivers access devices supported by the network adaptor through the device interface driver.

23.   (Currently Amended) The system of claim 14, wherein the devices determined ~~to be available~~ in the network adaptor for which device objects are generated comprise less than all the devices ~~supported by~~ included in the network adaptor.

24.   (Original) The system of claim 14, wherein the processor, when executing the device interface driver, is further enabled to:

signal the operating system of the changed configuration of devices available in the device hardware in response to detecting a change in the configuration of devices supported by the device hardware, wherein the operating system is capable of loading or unloading device drivers to support the changed configuration of devices available in the device hardware.

25.   (Currently Amended) The system of claim 14, wherein one device includes a protocol engine supporting the Internet Small Computer Interface (iSCSI) communication protocol and Transmission Control Protocol (TCP) communication protocol, wherein one device includes a protocol engine supporting an offloaded Local Area Network (LAN) communication protocol, and wherein one device includes a protocol engine supporting non-offloaded-LAN protocol.

26.   (Currently Amended) A system in communication with a network, comprising:

a processor;

an operating system executed by the processor;

a network adaptor supporting a plurality of devices included in the network adaptor;

at least one physical interface included in the network adaptor enabling communication of packets between the devices and the network, wherein the at least one physical interface is capable of interfacing with copper wires.

a device interface driver that represents the network adaptor as a virtual bus to the operating system, and that represents to the operating system each device supported in the device hardware as a device attached to the virtual bus, wherein the processor, when executing the device interface driver, is enabled to:

[[(i)]] access the network adaptor to determine the devices ~~supported by~~ included in the network adaptor;

[[(ii)]] generate one device object for each determined device ~~supported by~~ in the network adaptor, wherein each generated device object represents the determined device to the operating system; and

[[(iii)]] report the determined devices to the operating system, wherein the operating system loads a device driver for each of the reported devices supported by the network adaptor, and wherein to the operating system each determined device is considered attached to the virtual bus.

27. (Original) The system of claim 26, wherein the processor, when executing the device interface driver, is further enabled to:

report to the operating system that the determined devices are dependent on the virtual bus, wherein in response to being notified that the determined devices and virtual bus are dependent, the operating system will not remove the device interface driver representing the virtual bus until the device drivers associated with the determined devices are removed.

28. (Currently Amended) An article of manufacture comprising a computer readable storage medium including computer executable code for interfacing with device hardware supporting a plurality of devices included in the device hardware, wherein the ~~article of manufacture~~ code causes operations to be performed, the operations comprising:

initializing a device interface driver to represent the device hardware as a virtual bus to an operating system and to represent to the operating system each device supported in the device hardware as a device attached to the virtual bus;

initializing the device hardware;

accessing the device hardware to determine the devices ~~supported by~~ included in the device hardware;

generating one device object for each determined device ~~supported by~~ in the device hardware, wherein each generated device object represents the determined device to the operating system; and

reporting the determined devices to the operating system, wherein the operating system loads a device driver for each of the reported devices supported by the device hardware.

29. (Original) The article of manufacture of claim 28, wherein the operations further comprise:

reporting to the operating system that the determined devices are dependent on the virtual bus, wherein in response to being notified that the determined devices and virtual bus are dependent, the operating system will not remove the device interface driver representing the virtual bus until the device drivers associated with the determined devices are removed.

30.   (Original) The article of manufacture of claim 28, wherein the operations further comprise:

reporting to the operating system that a power state of the virtual bus represented by the device interface driver cannot be altered until all the device drivers representing devices attached to the virtual bus have their power state similarly altered.

31.   (Original) The article of manufacture of claim 28, wherein the hardware device comprises a network adaptor and wherein each device available in the network adaptor supports a protocol engine for different communication protocols.

32.   (Original) The article of manufacture of claim 31,, wherein each protocol engine processes packets according to a communication protocol and a network protocol, wherein each transport engine supports a different communication protocol and uses a same network protocol.

33.   (Original) The article of manufacture of claim 28, wherein the operations further comprise:

receiving a packet from one device driver;

determining a device queue in the device hardware queuing packets for the device supported by the device hardware corresponding to the device driver; and

writing the received packet to the determined queue.

34.   (Original) The article of manufacture of claim 28, wherein the operations further comprise:

receiving notification from the device hardware concerning transmission of one packet;

determining the device driver for the device in the network adaptor that processed the packet; and

transmitting notification to the determined device driver indicating the notification received from the device hardware.

35.   (Original) The article of manufacture of claim 28, wherein the operations further comprise:

receiving indication of a packet provided by the device hardware;

determining the device driver for the device supported by the network adaptor that processed the provided packet;

invoking a call to cause the determined device driver to process the provided packet.

36.   (Original) The article of manufacture of claim 28, wherein all the device drivers access devices supported by the device hardware through the device interface driver.

37.   (Currently Amended) The article of manufacture of claim 28, wherein the devices determined ~~to be available~~ in the device hardware for which device objects are generated comprise less than all the devices ~~supported by~~ included in the device hardware.

38.   (Original) The article of manufacture of claim 28, wherein the operations further comprise:

in response to detecting a change in the configuration of devices supported by the device hardware, signaling the operating system of the changed configuration of devices available in the device hardware, wherein the operating system is capable of loading or unloading device drivers to support the changed configuration of devices available in the device hardware.

39.   (Original) The article of manufacture of claim 28, wherein the operations of initializing the device hardware, accessing the device hardware to determine devices supported by the device hardware, generating the device objects, and reporting the determined devices to the operating system are performed by the device interface driver.

40.   (Currently Amended) The method of claim 28~~1~~, wherein one device includes a protocol engine supporting the Internet Small Computer Interface (iSCSI) communication

protocol and <u>Transmission Control Protocol (TCP</u><u>) communication</u> protocol, wherein one device includes a protocol engine supporting an offloaded <u>Local Area Network (LAN</u><u>) communication</u> protocol, and wherein one device includes a protocol engine supporting non-offloaded-LAN protocol.